# **3. Analízis modell kidolgozása**

## **3.1 Objektum katalógus**

### **3.1.1 Celhely**

A pálya egy eleme. Olyan üres mező, amely meg van jelölve. Ha jelölt ládát tolnak rá akkor a láda, és a mező is elveszti a jelölést, illetve a játékos, aki a ládát tolta pontot kap.

### **3.1.2 Csapdaajto**

A pálya egy eleme. Ha zárva van, akkor üres mezőként, ha nyitva, akkor lyukként viselkedik. Az állása a hozzákötött kapcsolótól függ.

### **3.1.3 Doboz**

Mozgatható elem. A játékosok ezeket tologatják a pályán. Alapterülete egy mezőével egyenlő és összenyomhatatlan. A dobozok képesek egymást vagy egy játékost eltolni, ha egy másik játékos rájuk tolja a dobozt. Ha a játékost nincs hova eltolni, azaz nincs mögötte üres mező, akkor a játékos meghal, és a doboz átveszi a helyét.

### **3.1.4 Fal**

A pálya egy eleme. Ez sem tartalmaz semmit. A játékosok nem képesek ládát tolni rá, vagy róla le, és nem képesek rálépni, vagy róla le. Ezek képzik majd az akadályokat a pályán.

**3.1.5 irany**

Egy enum, amivel az irányokat jelöljük majd.

### **3.1.6 Jatek**

A program fő része, ami kezeli, a felhasználó parancsait, és kliens működés esetén kommunikál a szerverrel.

### **3.1.7 Jatekos**

A raktári dolgozók, akik a dobozokat tolják a pályán. A felhasználó ezt irányítva tud játszani. A játék célja, hogy az összes jelölt dobozt a célhelyekre juttassuk.

### **3.1.8 JeloltDoboz**

Olyan doboz, amit a játékosnak a célhelyre kell tolnia. Ha rátolta, akkor a doboz és a célhely is elveszti a jelölést, és doboz valamint üres mező lesz belőlük. Minden más esetben dobozként viselkedik.

### **3.1.9 Kapcsolat**

Ez a szerver-kliens kapcsolatokat fogja reprezentálni szerveroldalon. Minden bejövő kapcsolat kap egy saját kommunikációs csatornát, amin a szerverrel kommunikálhat.

### **3.1.10 Kapcsolo**

A pálya egy eleme. Ha egy játékos egy ládát tol a kapcsolóra, akkor az kinyitja a hozzákapcsolt csapdaajtót. Ha játékos áll rajta, akkor nem csinál semmit.

### **3.1.11 KliensAdat**

A kliens számára szükséges adatokat tartalmazza, amik alapján ki tudja rajzolni a képernyőre az aktuális játékállást, pontokat. Ezek alkotják majd a szervertől a kliens felé mutató adatforgalmat.

### **3.1.12 Leptetheto**

Minden olyan objektumot reprezentál amik a pályán mozognak. Ide tartozik a játékos és a dobozok is.

### **3.1.13 Lyuk**

A pálya egy eleme. Olyan mező, amelybe lépve a játékos, vagy láda eltűnik.

### **3.1.14 OlvasoSz**

Ez az objektum fogja a bejövő adatforgalmat kezelni a szerver oldalán. Kiolvassa az adatokat az adatfolyamból.

### **3.1.15 OlvasoK**

Ez az objektum fogja a bejövő adatforgalmat kezelni a kliens oldalán. Kiolvassa az adatokat az adatfolyamból.

### **3.1.16 Palya**

A pálya tartalmazza a mezőket. A játékosok ezen az objektumon végzik el a teendőiket. Ez jegyzi meg a pálya állását, amit a játék a kliensekkel közöl, ez köti össze a játék elemeit. Számon tartja, és lépteti a játékosokat.

### **3.1.17 ParancsAdat**

A szerver számára szükséges adatokat tartalmazza, amik alapján léptetheti a játékost. Ezek alkotják majd a klienstől a szerver felé mutató adatforgalmat.

### **3.1.18 Pont**

A játékosok pontszámait tárolja el a játékosok azonosítóival.

### **3.1.19 Pontok**

A játékosok pontjait tárolja.

**3.1.20 Szerver**

A szerver a játék egyik központi eleme. Ezen keresztül történik a kommunikálás a kliensek, és a szerver játékosa felé.

### **3.1.21 UresMezo**

A pálya egy eleme. Ez a legegyszerűbb elem. A játékosok képesek ládát tolni rá, vagy róla le, és képesek rálépni, vagy róla le.

## 

## **3.2 Statikus struktúra diagramok**

## **3.3 Osztályok leírása**

**3.3.1 Celhely**

**⬤ Felelősség**

A pálya egy eleme. Olyan üres mező, amely meg van jelölve. Ha jelölt ládát tolnak rá akkor a láda, és a mező is elveszti a jelölést, illetve a játékos, aki a ládát tolta pontot kap.

**⬤ Ősosztályok**

UresMezo → Celhely

**⬤ Attribútumok**

* **jel:** Ha a mezőre jelölt láda kerül, elveszti a jelölését. Ez az attribútum ezt jelöli.

**⬤ Metódusok**

* **Fogad(jd: JeloltDoboz, j: Jatekos):** Az üres mező fogad függvényeit egészíti ki még egyel, ami jelölt dobozt vár, és a visszatérési értéke a lépés sikeressége. A játékos referencia ahhoz kell, amikor egy sor megtolt doboz végén egy jelölt doboz pontot akar adni a játékosnak.

**3.3.2 Csapdaajto**

**⬤ Felelősség**

A pálya egy eleme. Ha zárva van, akkor üres mezőként, ha nyitva, akkor lyukként viselkedik. Az állása a hozzákötött kapcsolótól függ, de kezdetben minden csapda ajtó zárva van.

**⬤ Ősosztályok**

UresMezo → Csapdaajto

**⬤ Attribútumok**

* **Nyitva:** A csapdaajtó állapota, ami megmondja, hogy csukva vagy nyitva van-e.

**⬤ Metódusok**

* **Nyit():** A jel értékét igazra állítja, amivel az ajtót nyitott állapotba teszi.
* **Csuk():** A jel értékét hamisra állítja, amivel az ajtót csukott állapotba teszi.
* **Fogad(j: Jatekos):** ha a csapdaajtó állapota nyitott, akkor a rákerülő játékos meghal (és igazzal tér vissza), ha az állapota csukott, akkor úgy viselkedik, mint az ősosztálya.
* **Fogad(d: Doboz, j: Jatekos):** ha a csapdaajtó állapota nyitott, akkor a rákerülő doboz meghal (és igazzal tér vissza), ha az állapota csukott, akkor úgy viselkedik, mint az ősosztálya.

**3.3.3 Doboz**

**⬤ Felelősség**

Mozgatható elem. A Dobozok reprezentációjáért felelős osztály. A játék végén a saját végellenőrzését is ez végzi.

**⬤ Ősosztályok**

Leptetheto → Doboz

**⬤ Metódusok**

* **Vege():** Ellenőrzi, hogy a doboz képes e még mozogni, vagy már nem lehet semerre sem tolni.
* **Halal():** Eltünteti a dobozt a játékból.

**3.3.4 Fal**

**⬤ Felelősség**

A pálya egy eleme. Ezek képzik az akadályokat a pályán.

**⬤ Ősosztályok**

UresMezo → Fal

**⬤ Metódusok**

* **Fogad(j: Jatekos):** az UresMezo Fogad metódusát írja felül. Elutasítja az érkeztetési kérést (hamissal tér vissza).
* **Fogad(d: Doboz, j: Jatekos):** az UresMezo Fogad metódusát írja felül. Elutasítja az érkeztetési kérést (hamissal tér vissza)csapdaasadsaad.

**3.3.5 irany**

**⬤ Felelősség**

Enum osztály, a mozgási irányokat tárolja.

**⬤ Attribútumok**

* **Fel**
* **Le**
* **Jobbra**
* **Balra**

**3.3.6 Jatek**

**⬤ Felelősség**

Ez az osztály felelős a program és a játékosok kezeléséért.

**⬤ Attribútumok**

* **nev:** Az aktuális játékos neve, amivel a játékban azonosítani lehet.
* **pontok:** A játékosok pontjai. A játék vége utáni kiiratáshoz kell.
* **kliensolvaso:** A kliens adatfolyam olvasó szálja, ami a hálózaton figyeli a szervertől érkező adatokat.
* **szerver:** A játék szervere.
* **parancs:** a játék kliensként parancsadatokat küld a szervernek.

**⬤ Metódusok**

* **Rajzol(k: KliensAdat):** kliens működés esetén a bejövő adatok alapján rajzolja ki a pályát.
* **Rajzol():** szerver működés esetén a meglévő adatokból rajzolja ki a pályát.
* **End():** A játék végén hívódik meg. Ezzel lehet lezárni a pályát.

**3.3.7 Jatekos**

**⬤ Felelősség**

A raktári dolgozók, akik a dobozokat tolják a pályán. A felhasználó ezt irányítva tud játszani.

A lépések megvalósításáért felelős.

**⬤ Ősosztályok**

Leptetheto → Jatekos

**⬤ Attribútumok**

* **nev:** A felhasználó azonosítója, aki a játékost irányítja.

**⬤ Metódusok**

* **Lepes(i: irany):** A pálya hívja meg, hogy léptesse a játékost, ami meghívja a mező fogad függvényét.
* **PontotKap():** A jelölt ládák célhelyre tolásakor a játékos pontot kap. Ez a függvény felelős érte.
* **Halal():** Megöli a játékost.

**3.3.8 JeloltDoboz**

**⬤ Felelősség**

Olyan doboz, amit a játékosnak a célhelyre kell tolnia. Ha rátolta, akkor a jelölt doboz elveszti a jelölést.

**⬤ Ősosztályok**

Leptetheto → Doboz → JeloltDoboz

**⬤ Attribútumok**

* **jel:** Ha a célmezőre jelölt láda kerül, a JeloltDoboz elveszti a jelölését. Ez az attribútum ezt jelöli.

**⬤ Metódusok**

* **JeletTorol():** Törli a jelölést a dobozról.

**3.3.9 Kapcsolat**

**⬤ Felelősség**

Ez a szerver-kliens kapcsolatokat fogja reprezentálni szerveroldalon. A kimenő forgalom kezeléséért felelős. Külön szálon fut.

**⬤ Attribútumok**

* **fut:** A szál futásához van rá szükség. Leállításkor hamis értékre állítjuk.
* **szerver:** A szerver referenciája, amihez a kapcsolat tartozik.
* **szerverolvaso:** A szerver adatfolyam olvasó szálja, ami a hálózaton figyeli a klienstől érkező adatokat.

**⬤ Metódusok**

* **Leptet(string nev, irany i):** Továbbítja az olvasótól érkező parancsot.
* **Quit():** Leállítja a szál futását a “fut” hamisra állításával.

**3.3.10 Kapcsolo**

**⬤ Felelősség**

A pálya egy eleme. Ha egy játékos egy ládát tol a kapcsolóra, akkor az kinyitja a hozzákapcsolt csapdaajtókat, ha ő lép a kapcsolóra, akkor bezárja.

**⬤ Ősosztályok**

UresMezo → Kapcsolo

**⬤ Attribútumok**

* **csapdak:** A kapcsolóhoz tartozó csapdák.

**⬤ Metódusok**

* **Fogad(jatekos j):** Ha játékos lép rá akkor becsukja a csapdaajtókat.
* **Fogad(Doboz d, Jatekos j):** Ha doboz lép rá akkor kinyitja azokat.

**3.3.11 KliensAdat**

**⬤ Felelősség**

A kliens adatainak tárolásáért felelős. Minden olyan adatot tárol, amire a kliensnek szüksége van, hogy megjelenítendő felületet kirajzolhassa.

**⬤ Attribútumok**

* **palya:** Int tömb, ahol egy speciális kódot használunk a mezők tárolására, és ezeket a kódokat használjuk hálózaton való kommunikáláshoz.
* **pontok:** A teljes ponttábla.

**⬤ Metódusok**

* **Palya():** Visszaadja az int tömböt, hogy ki lehessen belőle rajzolni a pályát.
* **Pontok():** visszaadja a pontokat.

**3.3.12 Leptetheto**

**⬤ Felelősség**

Absztrakt osztály, ami segítségével a játékosokat és a dobozokat is képesek vagyunk a mezők azonos attribútumában tárolni.

**⬤ Attribútumok**

* **palya:** Referencia a pályára. A mező lekérdezésekhez kell.
* **mezo:** Az aktuális mezőt tárolja, amin az objektum van.

**⬤ Metódusok**

* **Halal:** absztrakt függvény a leszármazottak halálához.
* **Utkozik(irany i, jatekos j):** Ha egy mozgatható objektumot doboz tol meg, akkor ez hívódik meg. A játékos referencia ahhoz kell, amikor egy sor megtolt doboz végén egy jelölt doboz pontot akar adni a játékosnak.
* **Tol(irany i, Jatekos j):** Ha egy mozgatható objektumot játékos tol meg, akkor ez hívódik meg. A játékos referencia ahhoz kell, amikor egy sor megtolt doboz végén egy jelölt doboz pontot akar adni a játékosnak.

**3.3.13 Lyuk**

**⬤ Felelősség**

Olyan mező, amelybe lépve a játékos, vagy láda eltűnik.

**⬤ Ősosztályok**

UresMezo → Lyuk

**⬤ Metódusok**

* **Fogad(Jatekos j):** Meghívja a játékos halál függvényét, és igazzal tér vissza.
* **Fogad(Doboz d, Jatekos j):** Meghívja a doboz halál függvényét, és igazzal tér vissza.

**3.3.14 OlvasoK**

**⬤ Felelősség**

A hálózaton beértkező adatok olvasása kliens oldalon, és azokban tárolt információk, és adatok továbbítása a feladata.

**⬤ Attribútumok**

* **fut:** A szál futásához van rá szükség. Leállításkor hamis értékre állítjuk.
* **jatek:** kliensként a játék referenciája, aminek a szervertől kapott adatokat továbbítja.

**⬤ Metódusok**

* **Quit():** Leállítja a szál futását a “fut” hamisra állításával.

**3.3.15 OlvasoSz**

**⬤ Felelősség**

A hálózaton beértkező adatok olvasása szerver oldalon, és azokban tárolt információk, és adatok továbbítása a feladata.

**⬤ Attribútumok**

* **fut:** A szál futásához van rá szükség. Leállításkor hamis értékre állítjuk.
* **kapcsolat:** szerverként a kapcsolat referenciája, ahova az olvaso a klienstől kapott adatokat továbbítja.

**⬤ Metódusok**

* **Quit():** Leállítja a szál futását a “fut” hamisra állításával.

**3.3.16 Palya**

**⬤ Felelősség**

Tárolja a mezőket, játékosokat, dobozokat. Összeköti a játék különálló részeit, és előállítja a kirajzoláshoz szükséges adatokat. Ez végzi a játék végellenőrzését.

**⬤ Attribútumok**

* **mezok:** Eltárolja a pályán lévő mezőket egy egy-dimenziós tömbben.
* **jatekosok:** Referencia a játékban lévő játékosokra.
* **dobozok:** Referencia a dobozokra.
* **pontok:** A játékosok pontjait tárolja.
* **jatekadatok:** Az aktuális játékállás adatai. Hálózati kommunikációra, és a játék kirajzolásához van rá szükség.

**⬤ Metódusok**

* **Leptet(String nev, irany i):** A kapott parancsot továbbítva lépteti a játékost.
* **Mezo(UresMezo u, irany i):** A játékos számára ad vissza referenciát a meglépni szándékozott mezőről.
* **Halal(Játékos j):** Törli az adott játékosra mutató referenciát.
* **Halál(Doboz d):** Törli az adott dobozra mutató referenciát.
* **Adatgyujtes():** Visszaadja a járék adatait.
* **PontotKap(string nev):** Pontot ad az adott játékosnak.
* **AdatFrissites():** Frissíti a játék adatait a kliense és a játék kirajzolása számára.
* **Vegellenorzes():** Ellenörzi, hogy van e még doboz ami képes lépni.

**3.3.17 ParancsAdat**

**⬤ Felelősség**

Játékos mozgásához szükséges adatokat tárolja. hálózati kommunikációnál van rá szükség.

**⬤ Attribútumok**

* **i:** Az irany enum megfelelő értéke.
* **nev:** A játékos neve, aki a parancsot küldte.

**⬤ Metódusok**

* **Irany():** Visszadja az irányt.
* **Nev():** Visszaadja a nevet.

**3.3.18 Pont**

**⬤ Felelősség**

A kiosztandó pont megvalósítása.

**⬤ Attribútumok**

* **nev:** A pont tulajdonosának neve.
* **pont:** A játékos pontszáma.

**3.3.19 Pontok**

**⬤ Felelősség**

A pontok kiosztásáért és tárolásáért felelős osztály.

**⬤ Attribútumok**

* **pontok:** A játékosok azonosítói, és a hozzájuk tartozó pontok.

**⬤ Metódusok**

* **PontotKap(nev: string):** Az adott játékosnak ad pontot.
* **getPont(int i):** Visszaadja az i-edik pontot Pont objektumként.
* **getHossz():** Visszaadja a pontok számát.

**3.3.20 Szerver**

**⬤ Felelősség**

Játék információit továbbítja a megfelelő helyekre.

**⬤ Attribútumok**

* **kapcsolatok:** A szerver kliens kapcsolatok listája.
* **aktualispalya:** Az éppen folyó játék pályájának referenciája.

**⬤ Metódusok**

* **Leptet(string nev, irany i):** Továbbítja a játékos és a kliensek parancsait.
* **Adatgyujtes():** A kirajzoláshoz szükséges adatokat gyűjti be a pályától.
* **Start(fajl: string):** Elindítja a játékot, a kiválasztott pályával. A lobby miatt van rá szükség.
* **Fut():** Visszaadja, hogy a játék elkezdődött-e már.

**3.3.21 UresMezo**

**⬤ Felelősség**

A játékosok képesek ládát tolni rá, vagy róla le, és képesek rálépni, vagy róla le.

**⬤ Attribútumok**

* **koord:** A mező pályán lévő koordinája.
* **leptetheto:** A pályán tartózkodó léptethető objektum referenciája.

**⬤ Metódusok**

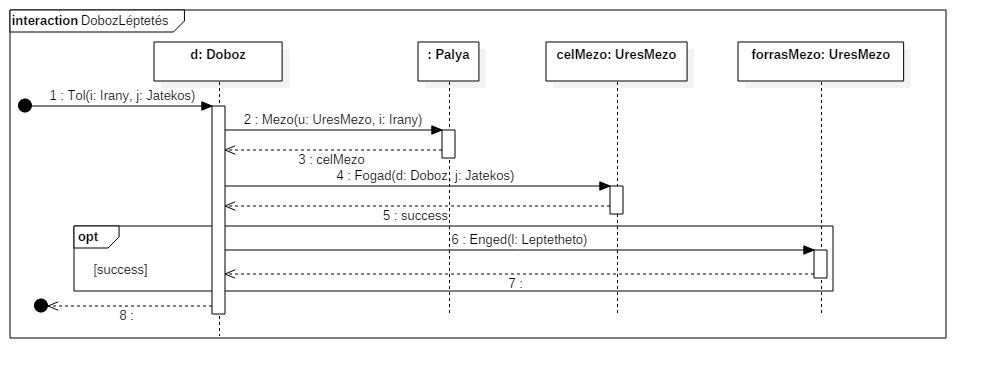
* **Fogad(Jatekos j):** Kezeli a játékos mezőre lépését.
* **Fogad(Doboz d, Jatekos j):** Kezeli a doboz mezőre tolását. A játékos referencia ahhoz kell, amikor egy sor megtolt doboz végén egy jelölt doboz pontot akar adni a játékosnak.
* **Enged(Leptetheto l):** Elengedi a léptethető objektumot, ha az más mezőre került.

## 

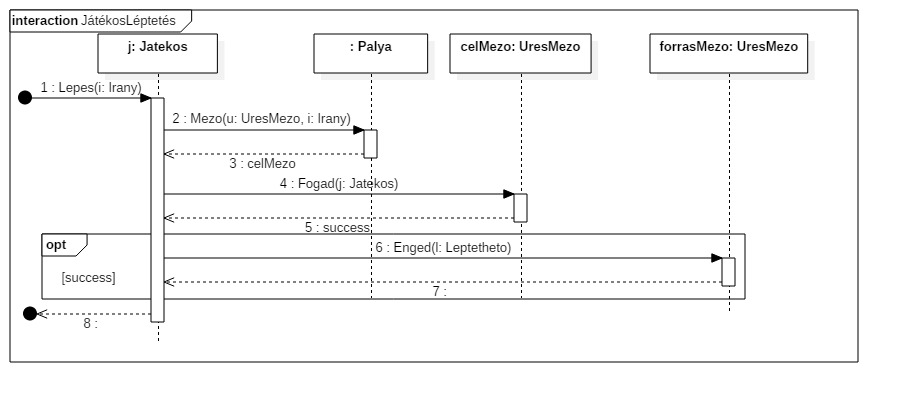
## **3.4 Szekvencia diagramok**

3.4.1 Játék vezérlő logikájának szekvencia diagramjai

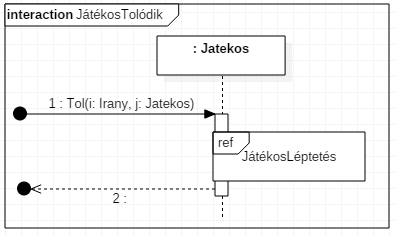
3.4.1.1 Doboz Léptetése



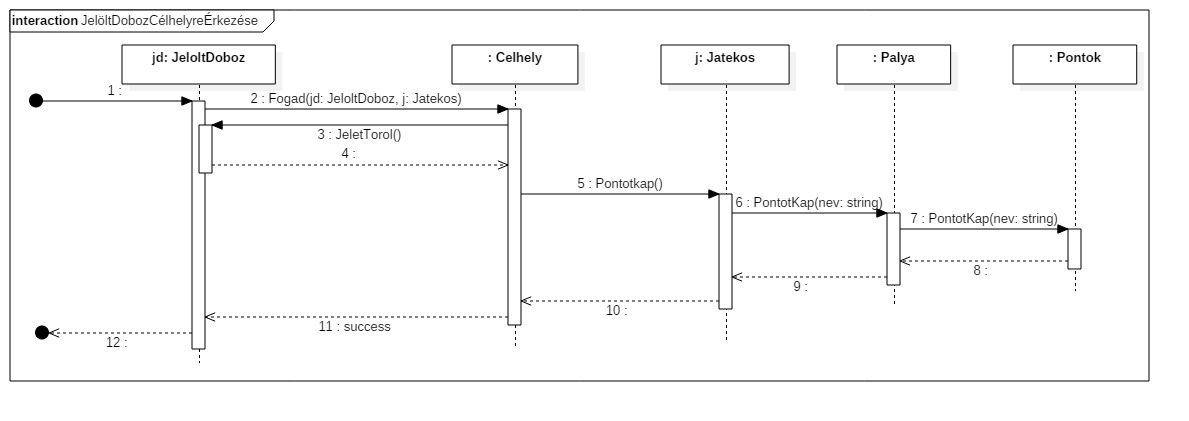
3.4.1.2 Játékos Léptetése



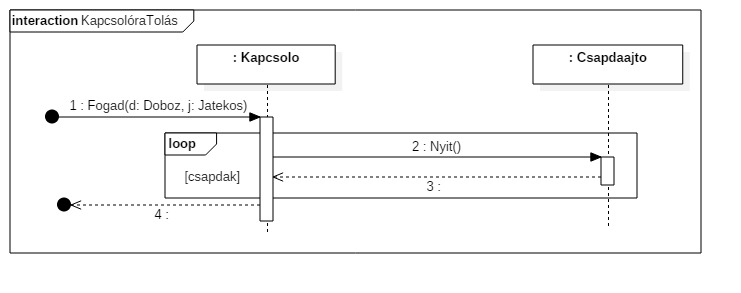
3.4.1.3 Játékos Tolódik



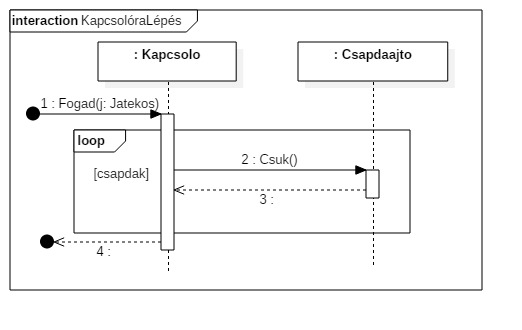
3.4.1.4 Jelölt Doboz Célhelyre Érkezése

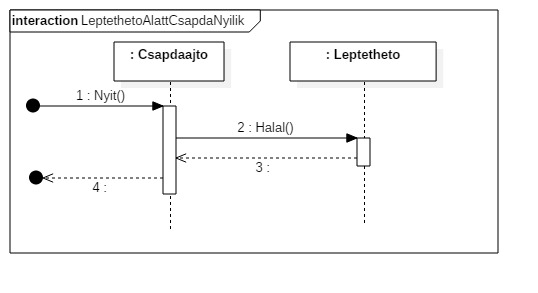


3.4.1.5 Kapcsolóra Tolás

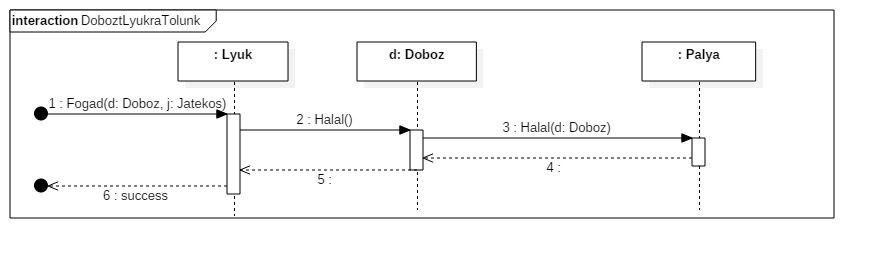


3.4.1.6 Kapcsolóra Lépés

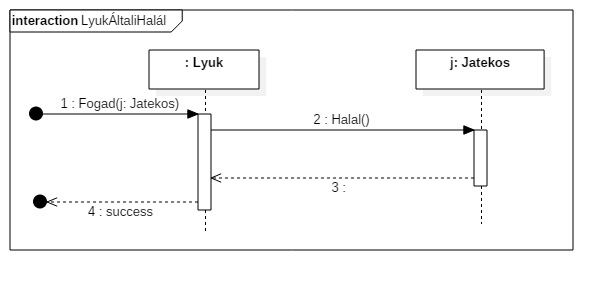


3.4.1.7 Léptethető Alatt Csapda Nyílik

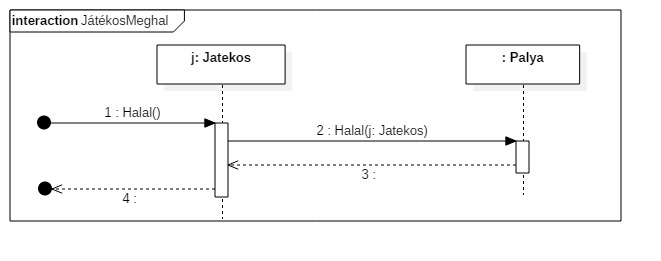
3.4.1.8 Dobozt Lyukra Tolunk



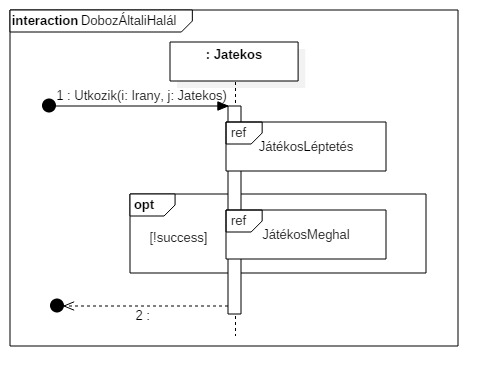
3.4.1.9 Játékos Lyukba Esik



3.4.1.10 Játékos Meghal

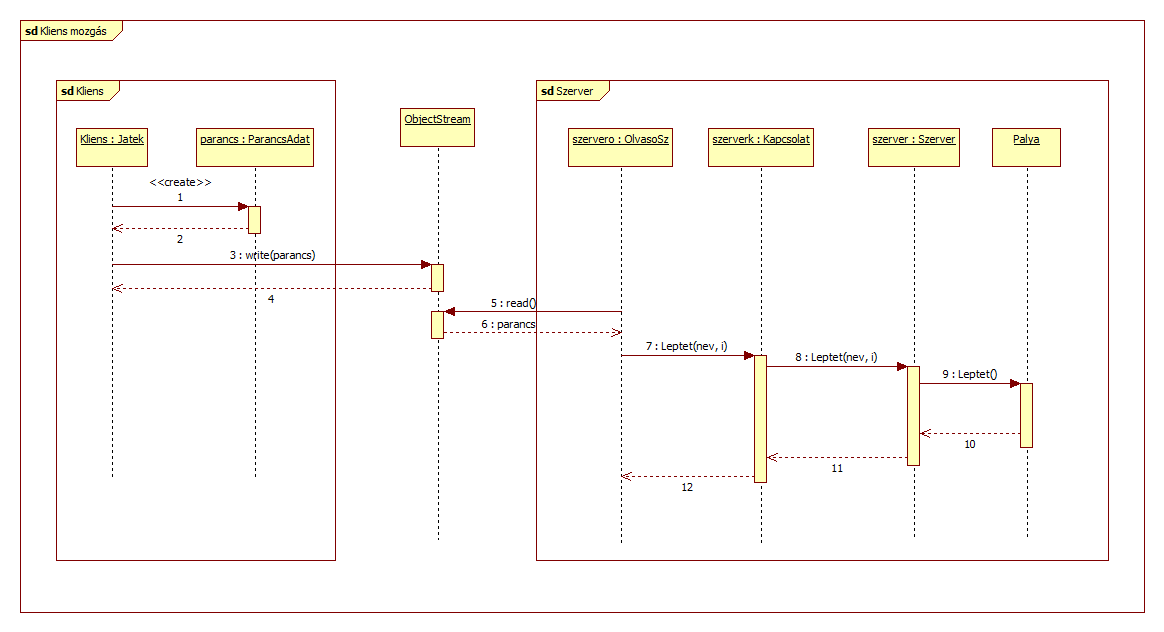


3.4.1.11 Játékost Doboz Össze Nyom

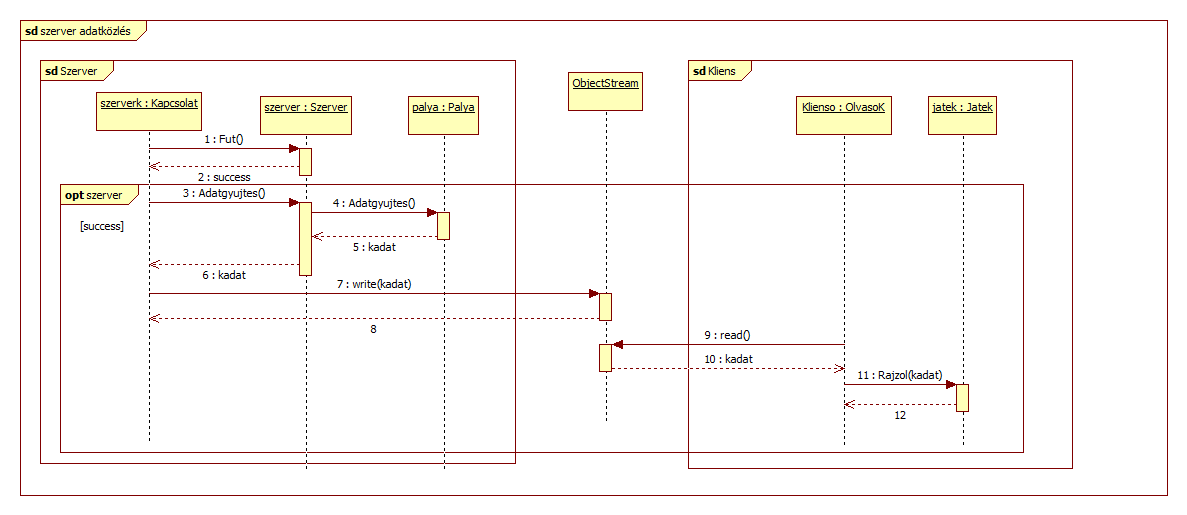


3.4.2 Szerver-Kliens kapcsolatot vezérlő szekvencia diagramok

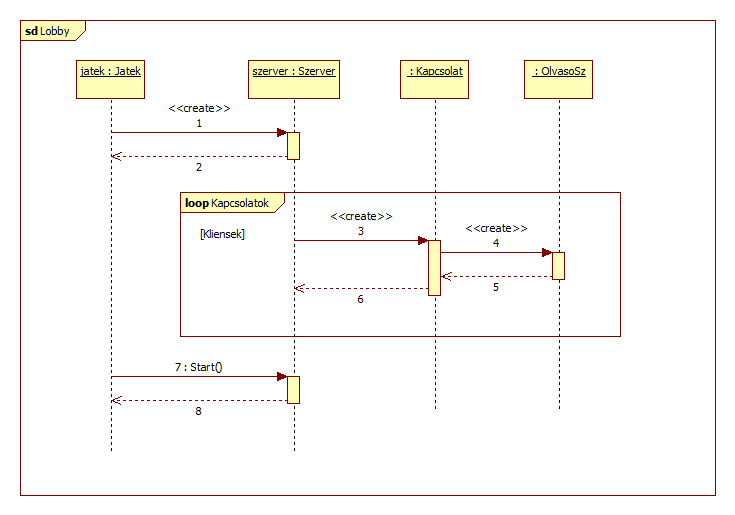
3.4.2.1 Kliens Mozgás



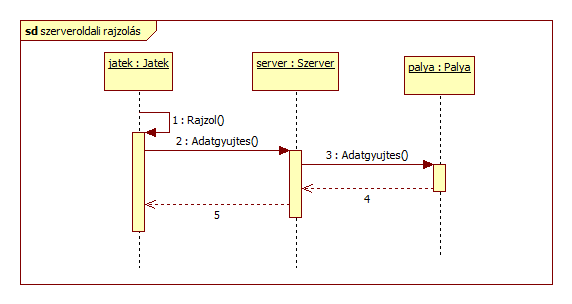
3.4.2.2 Szerver Adatközlés



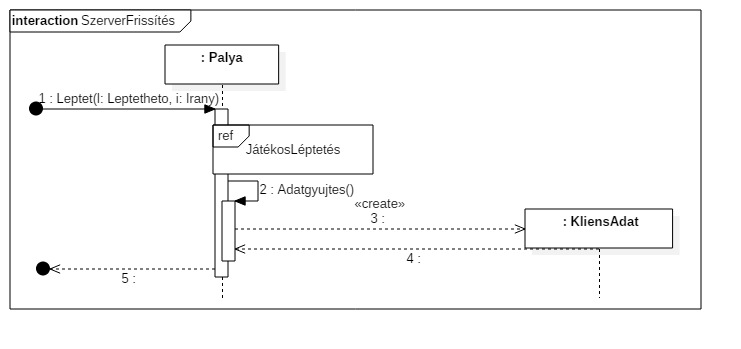
3.4.2.3 Lobby



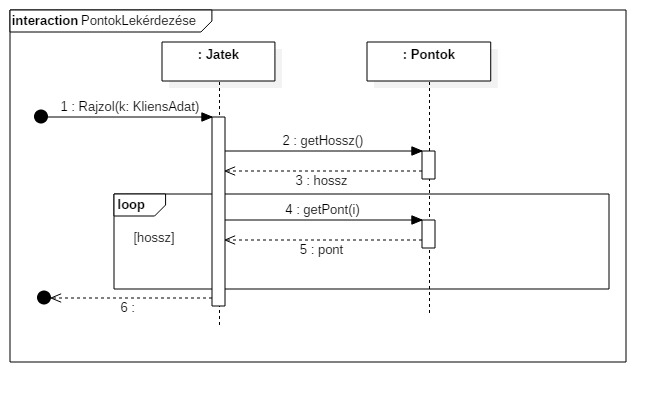
3.4.2.4 Szerveroldali Rajzolás



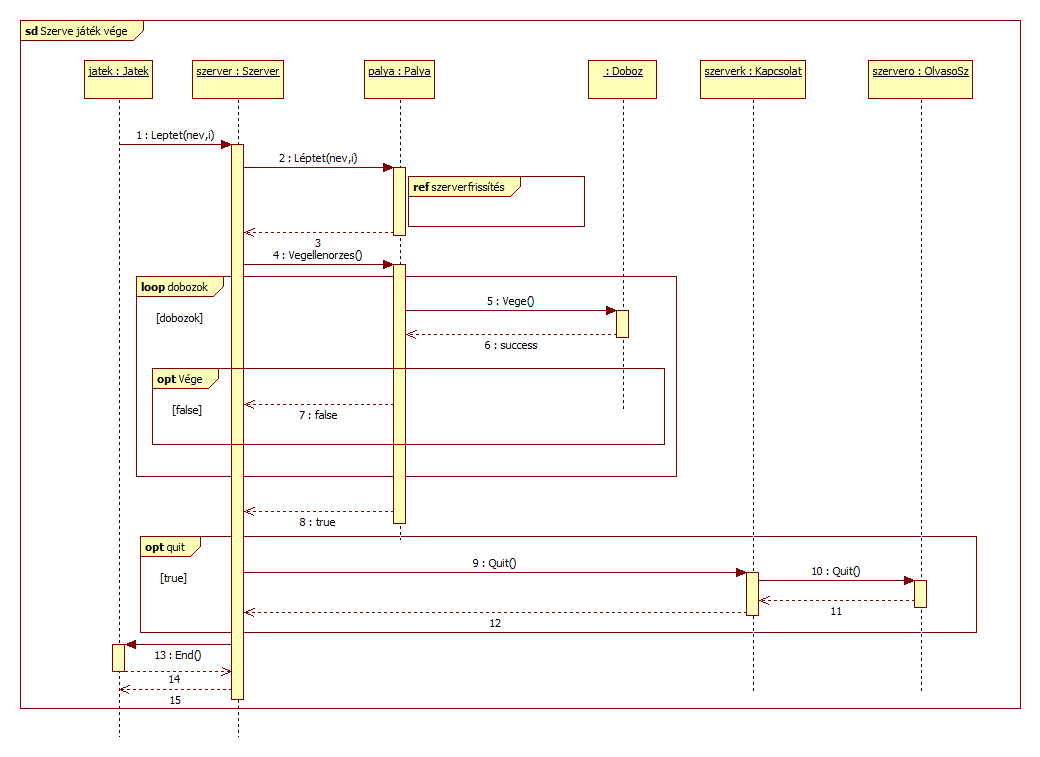
3.4.2.5 Szerver Frissítés



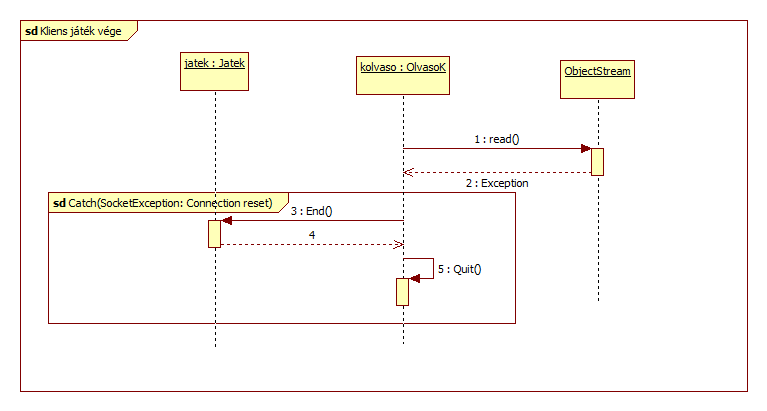
3.4.2.6 Pontok Lekérdezése



3.4.2.7 Szerver Játék Vége

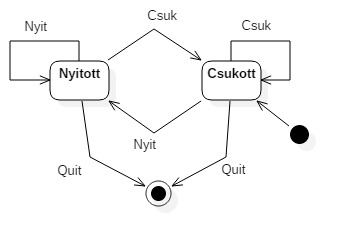


3.4.2.7 Kliens Játék Vége



## **3.5 State-chartok**

3.5.1 Csapdaajtók állása



## 

## **3.6 Napló**

|  |  |  |  |
| --- | --- | --- | --- |
| **Kezdet** | **Időtartam** | **Résztvevők** | **Leírás** |
| 2018.02.21. 8:00 | 1,5 óra | Bottlik  Holub  Horváth  Mocsári  Zsíros | Értekezlet.  Döntés: Zsíros elkészíti az osztálydiagramot. |
| 2018.02.21. 12:00 | 3,5 óra | Zsíros | Tevékenység:  Zsíros elkészíti az osztálydiagramot, és vázlatos leírást.  (3.1-3.3) |
| 2018.02.24. 13:30 | 2 óra | Mocsári | Szekvencia diagramok elkészítése |
| 2018.02.24. 15:30 | 11,5 óra | Mocsári  Zsíros | Tevékenység:  Mocsári és Zsíros újragondolja és elkészíti az osztálydiagramot, leírást, szekvencia diagramokat.  (3.1-3.5) |
| 2018.02.25. 12:10 | 3.5 óra | Bottlik Horváth  Mocsári  Zsíros | Dokumentum átnézése, kérdéses esetek megvitatása |